

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Bayu Endira, Dwi Husein, dkk., (2014) memanfaatkan *cloud computing* dalam aplikasi pencarian tambal ban terdekat. Aplikasi berjalan pada platform Android dengan memanfaatkan GPS. Aplikasi dapat mendeteksi dimana lokasi tambal ban terdekat dan menavigasikan lokasi tersebut dengan bantuan Google Maps yang berjalan secara cloud.

Eko Setiyawan (2009) meneliti tentang pemanfaatan layanan *Cloud PaaS* untuk penyimpanan dan sharing file menggunakan JSP. Pada penelitian tersebut dihasilkan sebuah aplikasi web yang memanfaatkan layanan cloud untuk kebutuhan sharing dan penyimpanan file. Aplikasi berjalan seluruhnya dengan mengandalkan aplikasi peramban (*browser*) dan ketersediaan akses internet.

Sahab Habibi (2010) meneliti tentang sistem informasi geografis berbasis web dalam pencarian lokasi wisata di daerah kabupaten Purworejo. Aplikasi yang dihasilkan tidak ditujukan

secara khusus untuk mobile dan belum memanfaatkan teknologi cloud.

Yang membedakan penelitian-penelitian tersebut dengan penelitian ini terletak pada implementasi teknologi *cloud* pada layanan *back-end* sebagai orientasi kebutuhan data pada aplikasi *Android native Java* untuk pencarian referensi lokasi *spot photo hunting*. API dikembangkan dengan bahasa PHP dan MySQL sebagai *database*. API menjadi jembatan untuk mengakses data-data yang tersimpan di *database* yang berjalan diatas *platform cloud*.

2.2. Dasar Teori

2.2.1. Cloud Computing

Komputasi awan (*cloud computing*) merupakan hasil kolaborasi antara pemanfaatan teknologi komputer dan teknologi internet dimana menurut IEEE komputasi awan adalah paradigma penyajian informasi secara permanen di komputer server dan tersimpan sementara di komputer klien. Komputasi awan direpresentasikan dalam bentuk awan karena hal ini mempermudah abstraksi infrastruktur yang sangat kompleks. Komputasi awan lebih cenderung menyajikan teknologi informasi

dalam bentuk layanan sehingga user cukup menyediakan piranti akses lengkap dengan internet untuk mendapatkan manfaatnya.

Teknologi cloud dapat dibagi ke dalam beberapa layer, yaitu :

a. Software as a Service (SaaS)

Konsep layanan yang memberikan kemudahan bagi pengguna untuk memanfaatkan sumber daya perangkat lunak yang sudah disiapkan provider dengan cara berlangganan sehingga tidak perlu mengeluarkan investasi yang terlalu besar. Namun demikian, SaaS hanya menyediakan fitur umum mengingat konsep ini bersifat *multi-tenant*.

b. Platform as a Service (PaaS)

Konsep layanan yang menyediakan modul-modul siap pakai dan dapat digunakan untuk mengembangkan sebuah aplikasi. Provider adalah pihak yang mengendalikan seluruh resource, sehingga pengguna layanan ini tidak memiliki kendali terhadap sumber daya komputasi dasar seperti memori, storage, *processing power*, dan lain-lain.

c. Infrastructure as a Service (IaaS)

Konsep layanan ini berada pada *the first tier layer*, sehingga setiap pengguna memiliki kontrol yang besar terhadap layanan yang disewanya. Layanan ini lebih ditujukan untuk konsumen dengan skala kebutuhan yang sangat tinggi dan kompleks dan lebih cenderung bersifat internal organization.

2.2.2. Teknologi Front-End dan Back-End

Aplikasi secara umum dapat dibedakan menjadi 2, yaitu front-end (client) dan back-end (server). Front-end adalah representasi akhir dari sebuah aplikasi yang biasanya langsung berhubungan langsung dengan user, sedangkan back-end adalah komponen yang mengatur tentang logika bisnis dari sebuah aplikasi.

Pengembangan front-end dan back-end pada sebuah aplikasi memiliki perbedaan baik ditingkat arsitektur dan lingkungan pengembangan seperti bahasa pemrograman. Teknologi front-end pada aplikasi web dapat dibangun dengan HTML, CSS dan juga Java Script, namun front-end pada aplikasi platform native yang jauh lebih spesifik dibangun dengan bahasa

Java, C#, Objective-C, dan lain sebagainya. Teknologi back-end merujuk pada seluruh teknologi yang ada pada sisi server, seperti database, pemrograman , web server, dan web service.

2.2.3. Cloud Backend as a Service

Cloud backend as a service merupakan pendekatan yang menyediakan alur keterhubungan antara web maupun mobile ke penyimpanan cloud. Pada umumnya layanan cloud backend memiliki fitur yang cukup lengkap seperti manajemen pengguna, *push notification*, integrasi media sosial, dan lainnya.

Cloud backend diciptakan untuk menjadi solusi pengembangan aplikasi secara lebih cepat, termonevitor, termanajemen dengan harga yang lebih rendah dari pada biasanya. Beberapa contoh penyedia yang terkenal diantaranya adalah AWS Web Services, Apache Usergrid, Appceletarator, Appery.io.

2.2.4. Spot Photo Hunting

Photo hunting merupakan sebuah aktifitas yang dilakukan baik individu maupun kelompok untuk mencari sebuah lokasi (spot) yang menarik dan menangkap momennya menggunakan kamera.

Spot memiliki beragam jenis, beberapa diantaranya dapat digunakan untuk memperkuat konsep sebuah acara dengan memberikan kesan yang kuat dari suasana yang diberikan spot. Aktivitas ini sering dilakukan mulai dari fotografer biasa, professional, maupun kelompok fotografi seperti DIY Fotografi.

2.2.5. Openshift Paas

OpenShift merupakan salah satu bentuk *Platform as a Service* (PaaS) yang disediakan oleh Red Hat untuk memenuhi kebutuhan komputasi awan bagi developer dan organisasi perusahaan dengan fitur *auto-scaling* dimana fitur ini akan menambahkan *resource* otomatis saat *traffic* besar terjadi dan menurunkan *resource* saat *traffic* kecil terjadi. *OpenShift* dapat digunakan secara khusus untuk mendeploy aplikasi menjadi cloud dengan dukungan bahasa pemrograman dan *framework* yang luas.

Dalam implementasinya, *OpenShift* menggunakan 2 komponen dasar, yaitu *Gears* dan *Cartridge*. *Gears* adalah *resource* yang dapat digunakan *Cartridge* berupa RAM dan alokasi penyimpanan. *Cartridge* menyediakan fungsionalitas untuk menjalankan aplikasi seperti MySQL, PHP, Ruby dan lain sebagainya.

OpenShift menyediakan 2 mekanisme pengaturan, yaitu *Management Console GUI* dan *Command Line*. Pengaturan dapat digunakan untuk *setting* dan manajemen akun, *launching* aplikasi, manajemen dan monitoring aplikasi.

2.2.6. Sistem Operasi Mobile Android

Android merupakan platform terbuka yang berjalan di berbagai perangkat device seperti smartphone, tablet, smart TV dan *vehicle auto*. Sistem operasi mobile pintar ini menjadi pemain utama disamping iOS dan Windows Phone. Dikembangkan oleh Google sejak November 2007 versi alpha dan hingga kini telah merelease versi 6.0 yang bernama "Marshmallow". Setiap major release, Android selalu dikeluarkan dengan kode nama yang spesifik :

- | | |
|----------------------------|-----------------------------|
| a. Cupcake (1.5) | g. Ice Cream Sandwich |
| b. Donut (1.6) | (4.0-4.0.4) |
| c. Eclair (2.0 – 2.1) | h. Jelly Bean (4.1 – 4.3.1) |
| d. Froyo (2.2 – 2.2.3) | i. KitKat (4.4-4.4W.2) |
| e. Gingerbread (2.3-2.3.7) | j. Lollipop (5.0-5.1.1) |
| f. Honeycomb (3.0 – 3.2.6) | k. Marshmallow (6.0-6.0.1) |

Tingginya API pada versi Android menjadi parameter acuan dalam pemuktahiran pengembangan teknologi baik dari sisi fitur, UI/UX, aksesori dan juga performa yang diimplementasikan pada sistem operasi mobile. Oleh karena itu Google memberikan dukungan kepada para pengembang aplikasi untuk menjembatani penggunaan fitur API terbaru agar dapat dijalankan pada API lama dalam bentuk *Application Compatibility* (AppCompat).

Aplikasi yang dibangun pada sistem operasi Android dikembangkan dengan bahasa Java dan IDE Android Studio, sedangkan dalam proses pembuatan APK kini didasarkan dengan konsep Gradle build.

2.2.7. Gradle

Gradle merupakan salah satu *build automation system* yang kini turut digunakan Google dalam proses pembangunan aplikasi Android. Gradle dikembangkan berdasarkan konsep Apache Ant dan Apache Maven namun berbasis *domain-specific language* (dsl) yang berbasis Groovy.

Gradle mendukung *incremental builds* dan juga mampu menentukan bagian mana yang *up-to-date*, sehingga kompilasi hanya akan dikerjakan pada bagian yang spesifik. Selain itu

Gradle mempermudah penyamaan lingkungan pengembangan seperti *dependency*, *build-tools*, dan juga kelengkapan lainnya. Hal ini sangat membantu dalam proses kolaborasi pekerjaan karena menjamin bahwa pengembang menggunakan *tools* yang sama dalam prosesnya.

Gradle dapat digunakan sebagai alat konfigurasi dan dalam implementasinya dituliskan dalam sebuah script bernama '*build.gradle*' secara modular. Setiap level modul memiliki script konfigurasi secara terpisah, sedangkan di Android, script utama sebagai konfigurator terletak pada '*build.gradle*' bagian terluar atau *toop-root*. Hingga kini, Gradle telah mencapai versi 3.0.0.

2.2.8. Google Map API

Google Map API merupakan *web-based service* yang memberikan layanan untuk para pengembang menanamkan map ke dalam aplikasi yang dikembangkan. Google Map memberikan beberapa pilihan untuk mendapatkan informasi sesuai kebutuhan, diantaranya pilihan citra satelit, jalan, lokasi dan arah.

Pada tahun 2005, Google memperkenalkan Google Map untuk perangkat mobile yang menargetkan untuk dijalankan pada platform *java-based phone* termasuk Android. Pada

peluncuran tersebut dikenalkan 2 produk yang berbeda, yaitu aplikasi Google Map untuk Android dan juga API untuk berkolaborasi dengan aplikasi lain.

Pada implementasinya, Android dapat berhubungan dengan Google Map melalui API yang disediakan melalui SDK pengembangan. Setiap aplikasi Android diidentifikasi dengan *application Id* yang tidak mungkin sama dengan aplikasi Android lain. *Application Id* didaftarkan ke dalam console google map untuk mendapatkan kunci otorisasi, sehingga data map yang ditampilkan tidak akan tertukar dengan aplikasi lainnya.

Google Map mampu menampilkan dimana lokasi pengguna berada dan juga pencarian lokasi lain sesuai kebutuhan dalam bentuk marker dan informasi. Selain itu, Google melengkapinya dengan tambahan navigasi dan tracking GPS.

2.2.9. PHP

PHP (*PHP Hypertext Preprocessor*) adalah bahasa script yang berjalan disisi server untuk kebutuhan pembangunan web dan *general-purpose programming language*. Bahasa ini merupakan bahasa yang dapat dipelajari secara bebas dan

digunakan secara meluas dibawah lisensi sumber terbuka (open source).

PHP sebagai *script language*, membutuhkan interpreter yang biasanya diimplementasikan dalam bentuk module di web server atau biasa disebut dengan *Common Gateway Interface* (CGI). Mesin web server mengkombinasikan hasil dari intepretasi tersebut dengan executed PHP code.

Pada tanggal 3 Desember 2015, PHP telah direlease hingga versi 7.0.0 dan seiring dengan perkembangan, PHP telah mendukung konsep *Object Oriented Model*. PHP memiliki dukungan yang sangat baik untuk berbagai macam jenis database seperti MySQL, MariaDB, Oracle dan PostgreSQL. PHP juga dapat dijalankan diberbagai web server sperti Apache, Lighttpd dan IIS.

PHP dapat dijadikan sebagai bahasa yang handal untuk dikolaborasikan sebagai bahasa pembangun service API aplikasi mobile. Namun demikian, logika pengamanan transaksi data harus menjadi perhatian utama mengingat kode PHP dibaca semua orang.

2.2.10. MySQL

MySQL dibangun dengan bahasa C/C++ dan merupakan salah satu dari keluarga *Relational Database Management System* (RDBMS) open source yang memiliki *full-featured* database manajemen system. Versi release terakhir dipublikasikan pada Oktober 2015 dan mencapai versi MySQL 5.7.

Beberapa keunggulan MySQL diantaranya :

- a. *Portable, Open Source* dan *Multi user support*.
- b. Mendukung tipe data yang sangat beragam.
- c. Dikembangkan dengan aspek keamanan yang baik.
- d. Didukung dengan Administrative Tools yang lengkap.
- e. Dapat diintegrasikan dengan berbagai bahasa pemrograman.

Hal yang menarik dari MySQL adalah dapat dijalankan pada platform berbasis Cloud Computing seperti pada Amazon EC2 sebagai Virtual Machine Image, *MySQL as a Service* atau MySQL Cloud Hosting.